FunG - Biomechanical Models in Modern C++

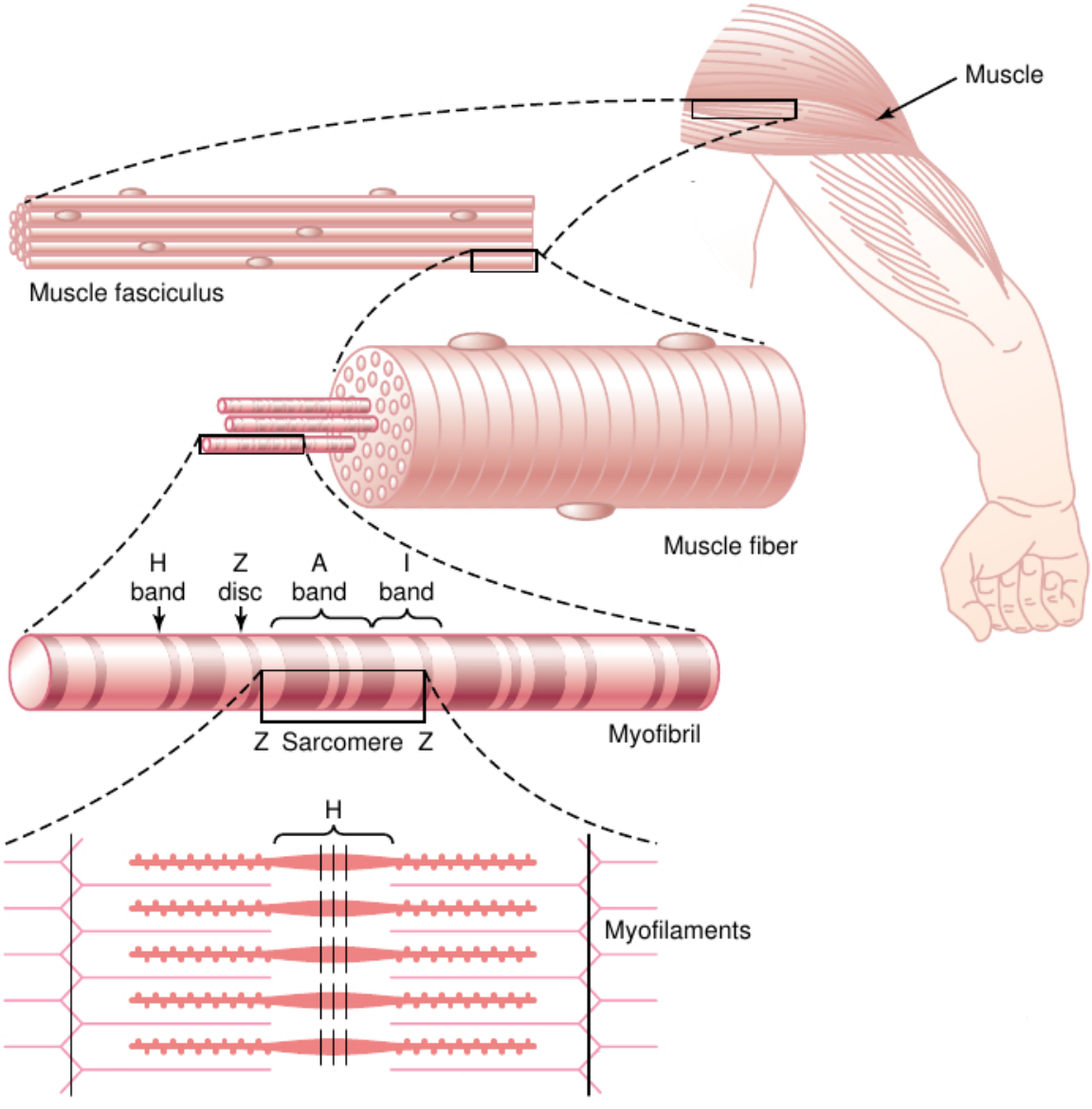
Lars Lubkoll

Yuan-Cheng Fung

- Founder of “Modern” Biomechanics

Yuan-Cheng Fung

- Founder of “Modern” Biomechanics
- Fung’s law: $\frac{\partial \sigma}{\partial C} = a\sigma$


Muscle
Muscle fasciculus
Muscle fiber
H band
Z disc
A band
I band
Z Sarcomere Z
Myofibril
H
Myofilaments

Half a model for muscle tissue

$$
\begin{aligned}
W(F) &= c\,[\exp(b(\bar{\iota}_1(C) - 3)) - 1] \\
&\quad + d\left[\exp(a(\bar{\iota}_6(C, M) - 1)^2) - 1\right] \\
\bar{\iota}_1 &= \operatorname{tr}(C)\det(C)^{-1/3}, \qquad C = F^T F \\
\bar{\iota}_6 &= \operatorname{tr}(CM^2)\det(C)^{-1/3}
\end{aligned}
$$

F : deformation gradient
M : structural tensor

Half a model for muscle tissue

```
auto model = c*( exp(b*(mi1(F)-3)) - 1 )
           + d*( exp(a*(mi6(F,M)-1)^2) - 1 );
auto W = finalize( model(strainTensor(F)) );
```

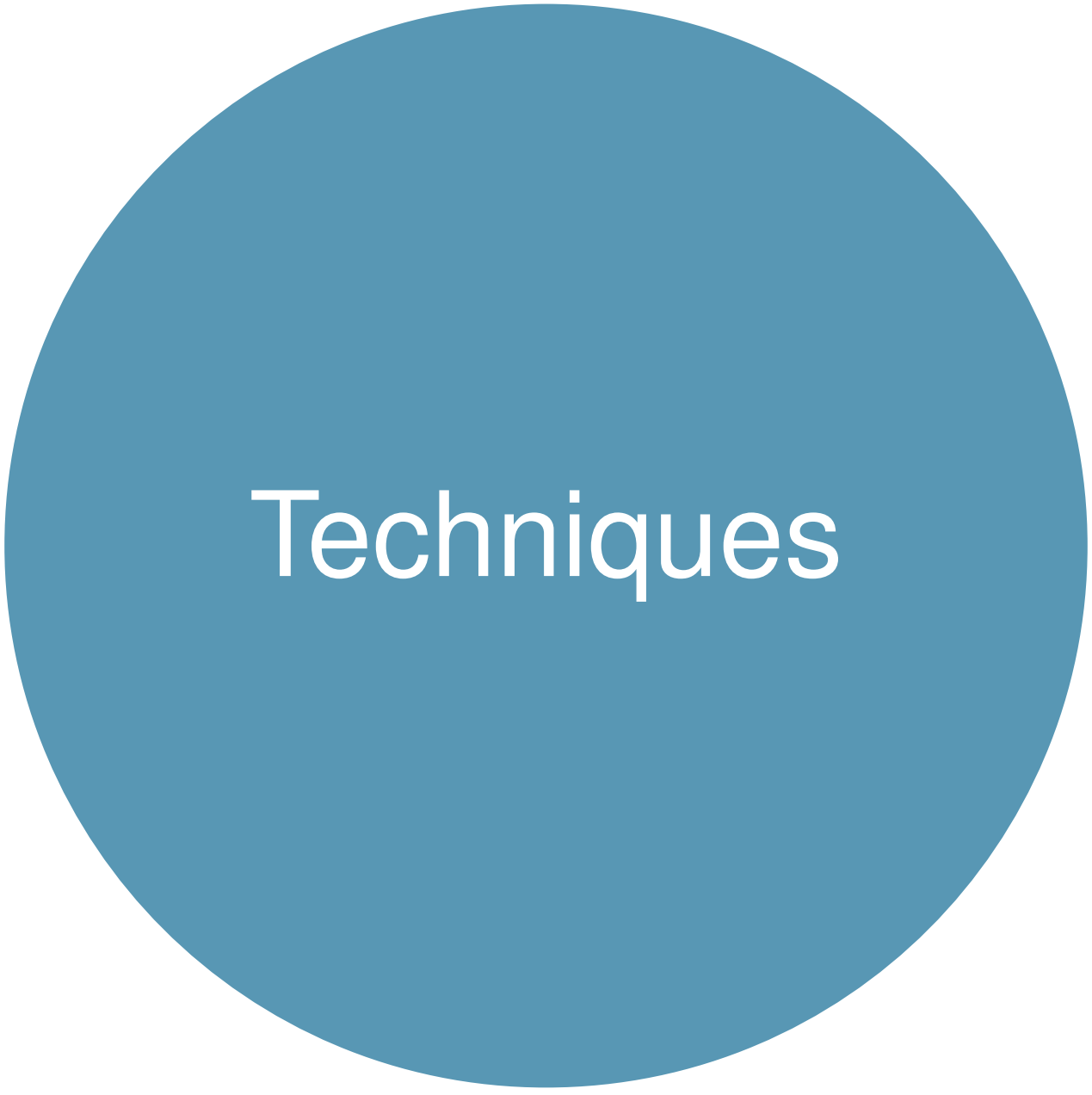
`F:` deformation gradient

`M:` structural tensor

Usage (given matrices, F, dFi):

```
// set function argument
 W.update(F);

// access function value and derivatives
 auto value = W(); // or W.d0()
 auto df1   = W.d1(dF0);
 auto df2   = W.d2(dF0,dF1);
 auto df3   = W.d3(dF0,dF1,dF2);
```


Techniques

Expression Templates + Operator Overloading

Expression Templates + Operator Overloading

- template programming (almost) without templates

Expression Templates + Operator Overloading

- template programming (almost) without templates
- enhance inlining capabilities

```
template <class Arg, int n = dim<Arg>()>
auto mi1(const Arg& x)
{
 return i1(x)*pow<-1,n>( det(x) );
}
```

Generic Programming

Generic Programming

- no types:
  → `auto, decltype, std::declval`

Generic Programming

- no types:
  $\rightarrow$ `auto, decltype, std::declval`
- (relaxed) vector space structure for user-defined types

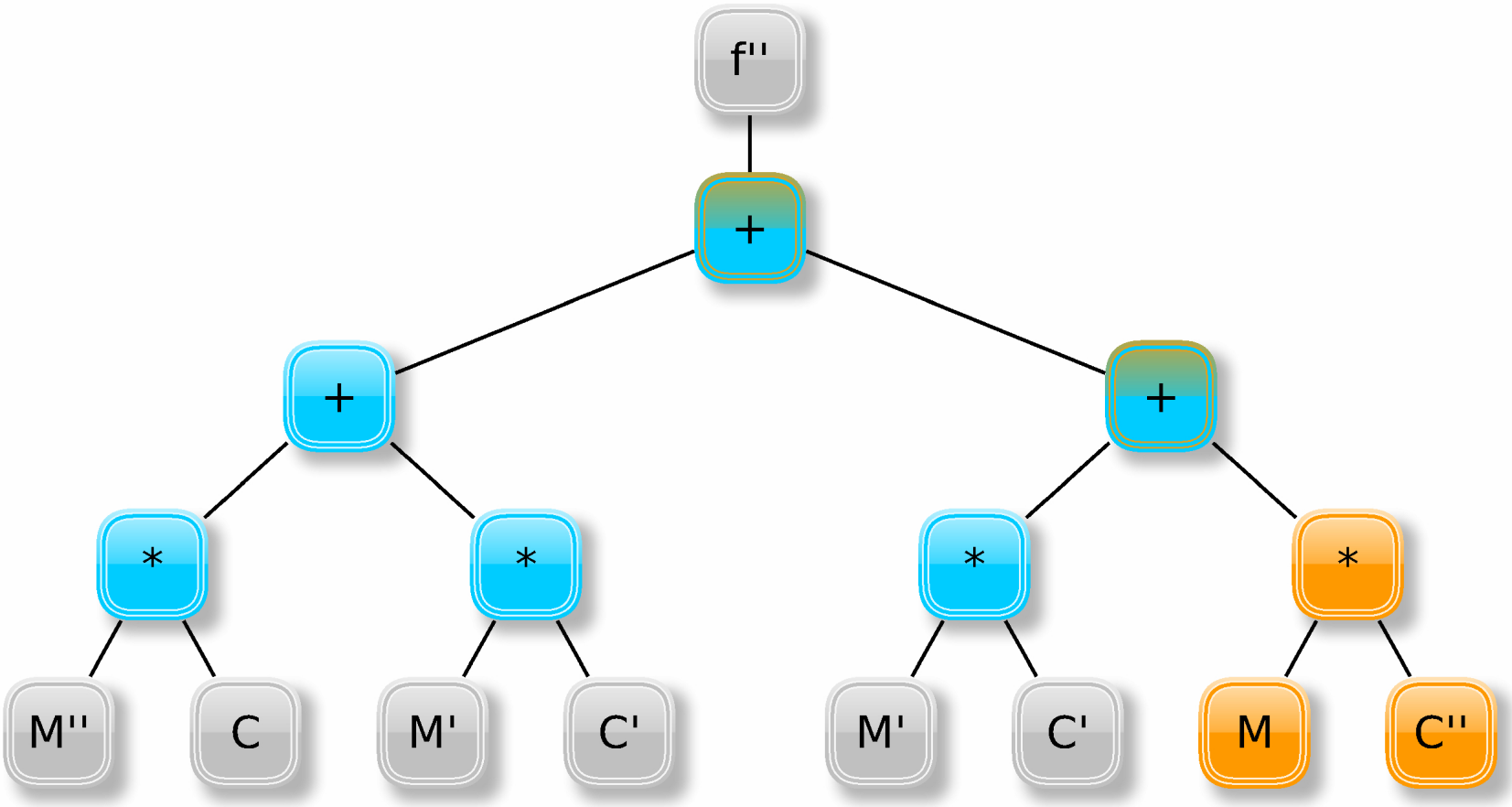
Generic Programming

- no types:
  $\rightarrow$ `auto, decltype, std::declval`
- (relaxed) vector space structure for user-defined types
- SFINAE-adjustments for Armadillo, Eigen, Dune::FieldMatrix


Optimization


Caching

Elimination of zeros ($\iota_4(F^TF, M) = \mathrm{tr}(f)$, $f = F^TFM$)



Speak with the compiler:

Speak with the compiler:

- max-inline-insns-auto = 5000

Speak with the compiler:

- max-inline-insns-auto = 5000
- early-inlining-insns = 5000

Speak with the compiler:

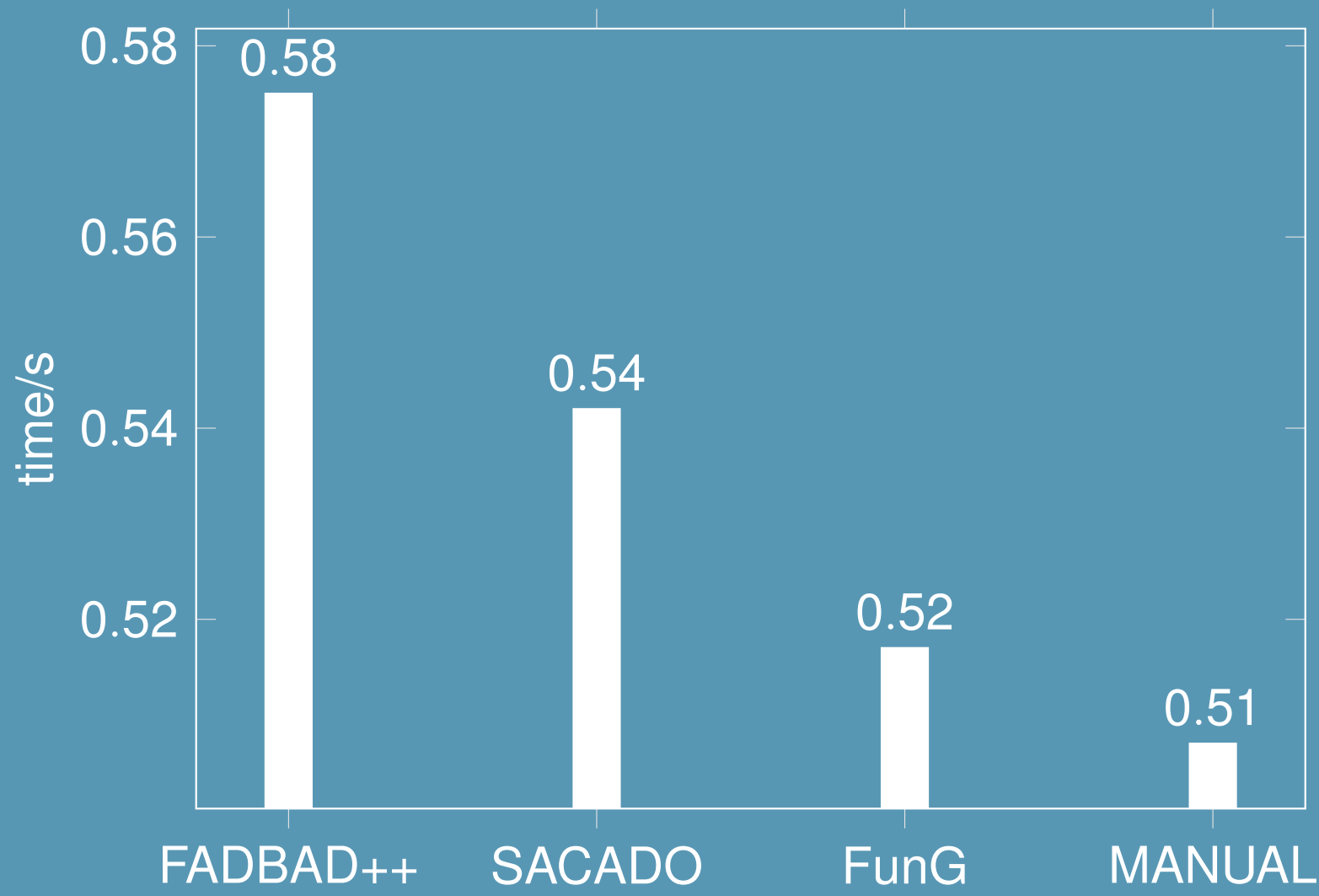
- max-inline-insns-auto = 5000
- early-inlining-insns = 5000
- inline-unit-growth = 100

Factor: up to 14x (av.: 2.5x)


Performance

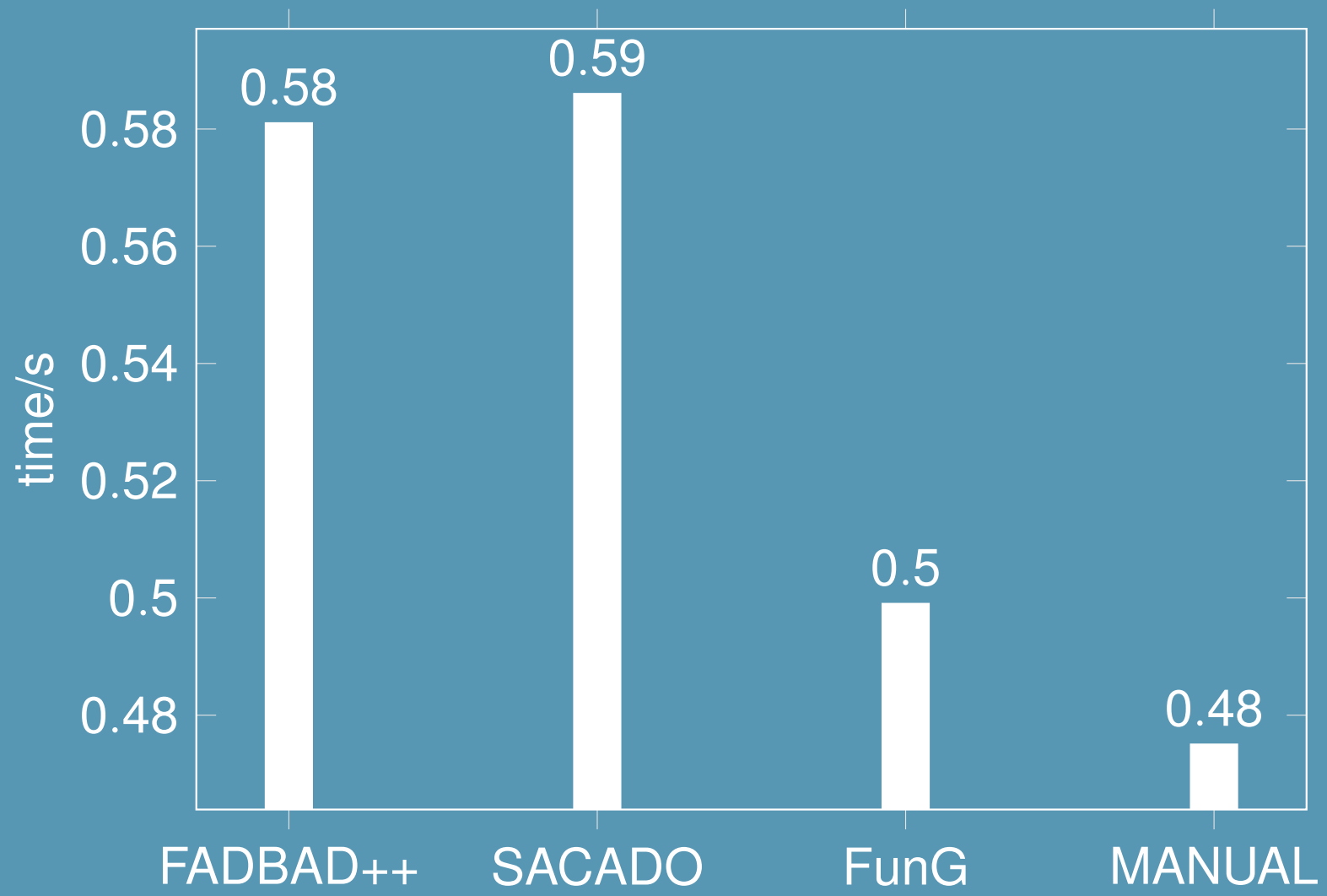
Comparison with AD libraries

$f(x) = x^{3/2} + \sin(\sqrt{x})$ (10^7 eval.)



Comparison with AD libraries

$f(x, y, z) = (y + z)\sqrt{x} + \sin(\sqrt{x})$ (10^7 eval.)



Reliability

Reliability

- unit and integration tests

Reliability

- unit and integration tests
- compile-time concept checks

Interested? http://lubkoll.github.io/FunG

Contact: lars.lubkoll@posteo.de

Half a model for muscle tissue

$W(F) = c\,[\exp(b(\bar{\iota}_1 - 3)) - 1] + d\,\left[\exp(a(\bar{\iota}_6 - 1)^2) - 1\right]$ (10^7 eval.)